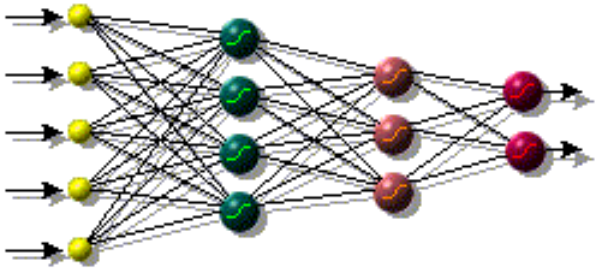
COMP 308 ARTIFICIAL INTELLIGENCE PART 8.2 – ARTIFICIAL NEURAL NETWORKS

Njeri Ireri

Jan – April 2020

Overview

- The Brain
- Brain vs. Computers
- The Perceptron
- Multilayer networks
- Some Applications

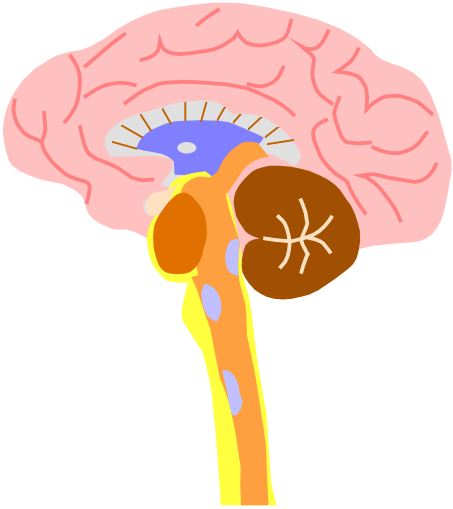


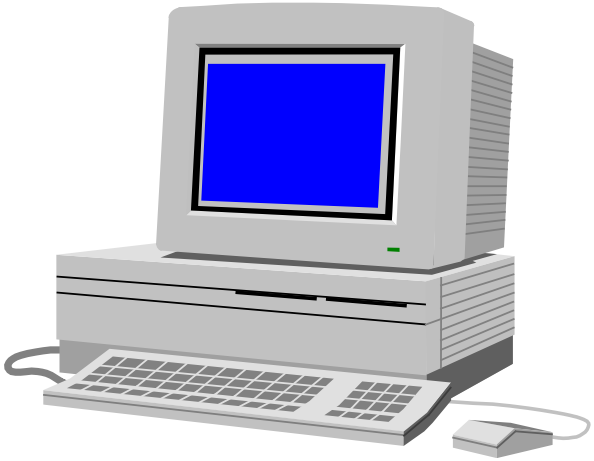
Artificial Neural Networks

- Other terms/names
  - connectionist
  - parallel distributed processing
  - neural computation
  - adaptive networks..
- History
  - 1943-McCulloch & Pitts are generally recognised as the designers of the first neural network
  - 1949-First learning rule
  - 1969-Minsky & Papert - perceptron limitation - Death of ANN
  - 1980's - Re-emergence of ANN - multi-layer networks

Brain and Machine

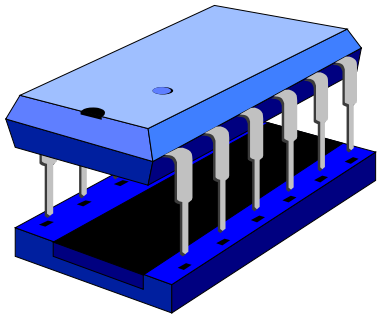
- The Brain
  - Pattern Recognition
  - Association
  - Complexity
  - Noise Tolerance





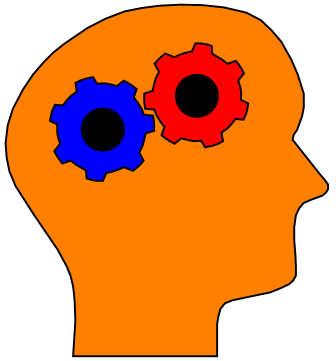
- The Machine
  - Calculation
  - Precision
  - Logic

The contrast in architecture



- The **Von Neumann architecture** uses a single processing unit;
  - Tens of millions of operations per second
  - Absolute arithmetic precision

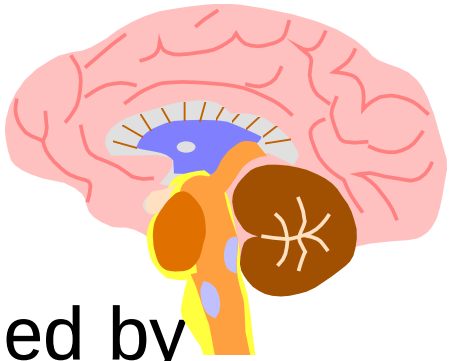
- **The brain** uses many slow unreliable processors acting in parallel



Features of the Brain

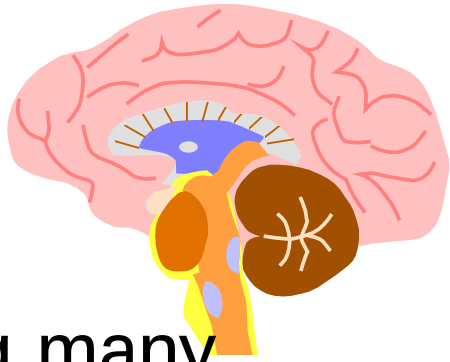
- Ten billion (10^{10}) neurons
- On average, several thousand connections
- Hundreds of operations per second
- Die off frequently (never replaced)
- Compensates for problems by massive parallelism

The biological inspiration



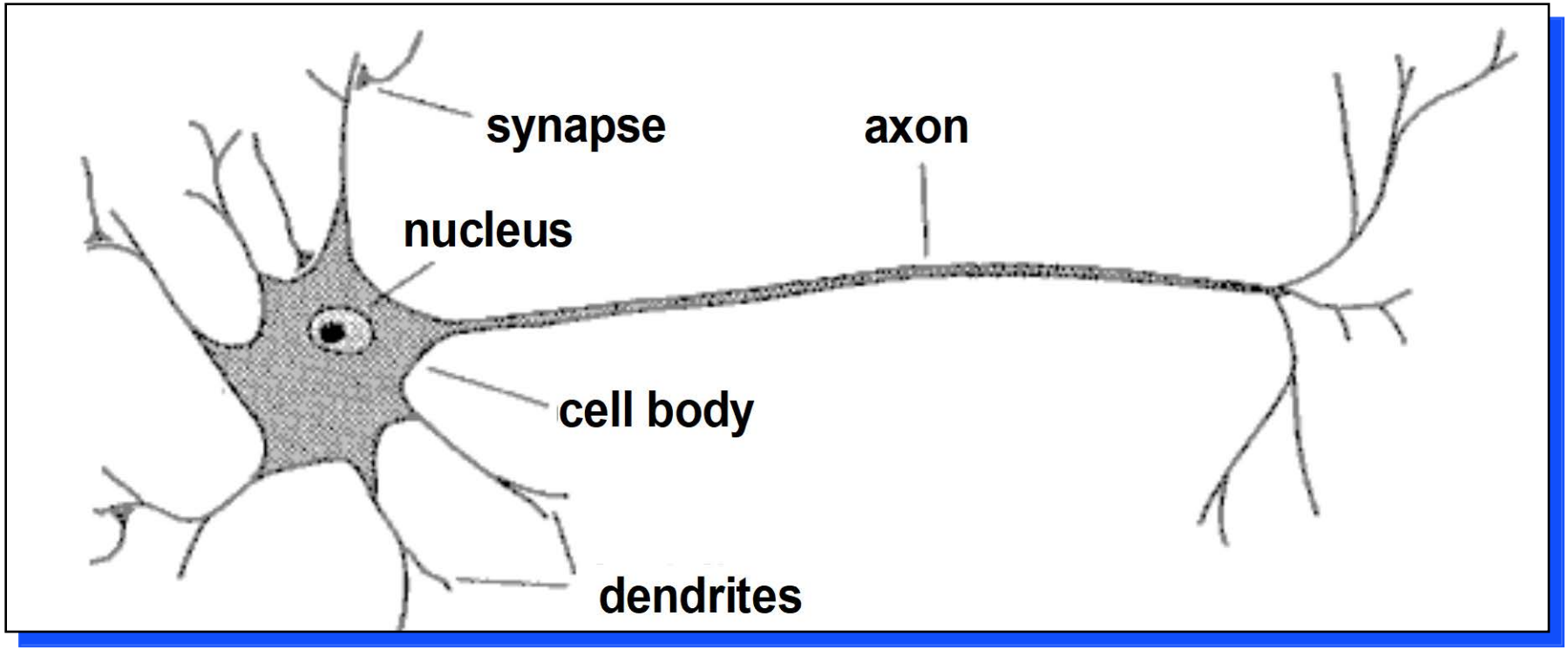
- The brain has been extensively studied by scientists.
- Vast complexity prevents all but rudimentary understanding.
- Even the behaviour of an individual neuron is extremely complex

The biological inspiration



- Single “percepts” distributed among many neurons
- Localized parts of the brain are responsible for certain well-defined functions (e.g. vision, motion).
- Which features are integral to the brain's performance?
- Which are incidentals imposed by the fact of biology?

The Structure of Neurons



The Structure of Neurons

The Structure of Neurons

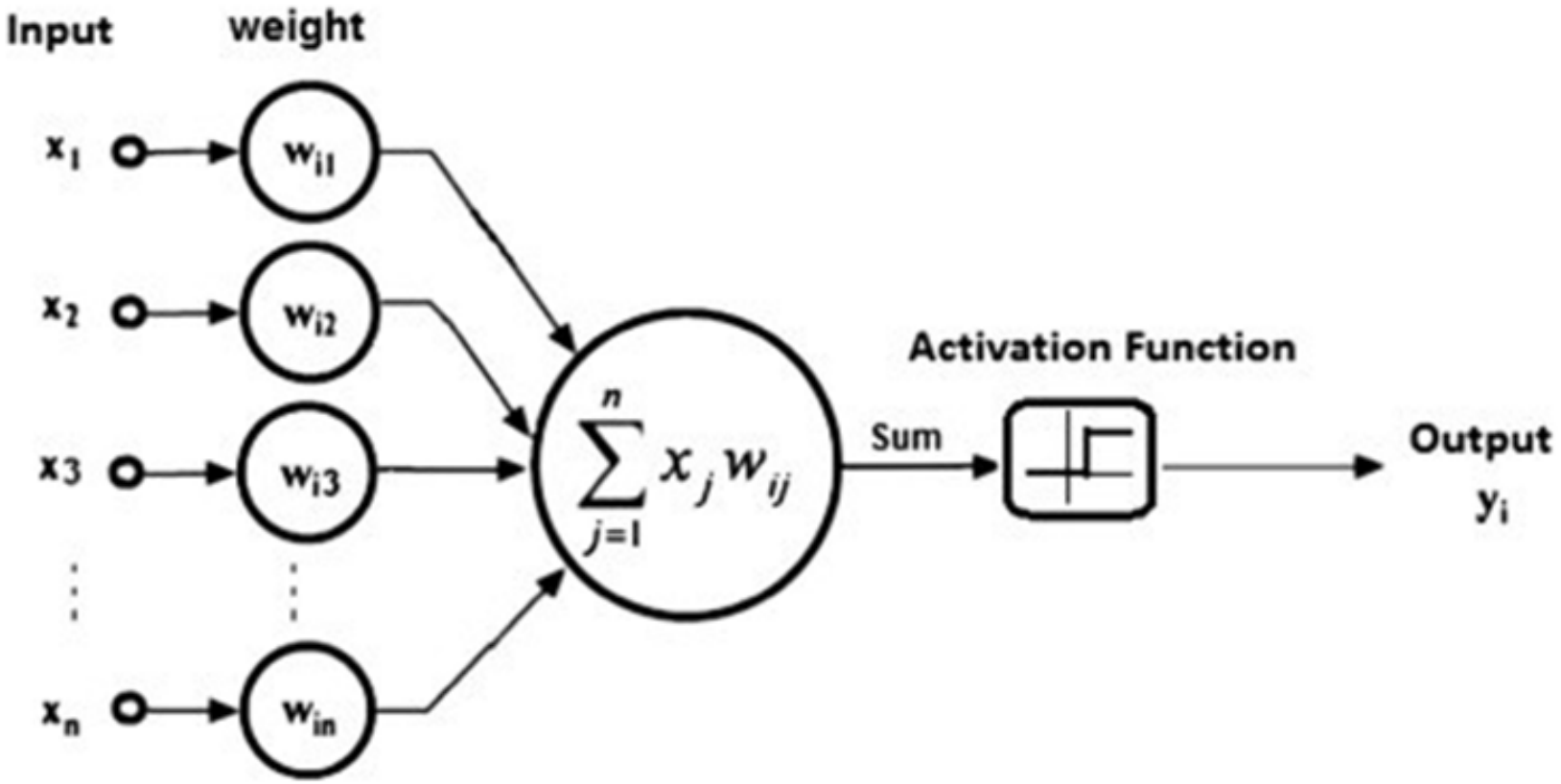
A neuron has a cell body, a branching input structure (the dendrIte) and a branching output structure (the axOn)

- Axons connect to dendrites via synapses
- Electro-chemical signals are propagated from the dendritic input, through the cell body, and down the axon to other neurons

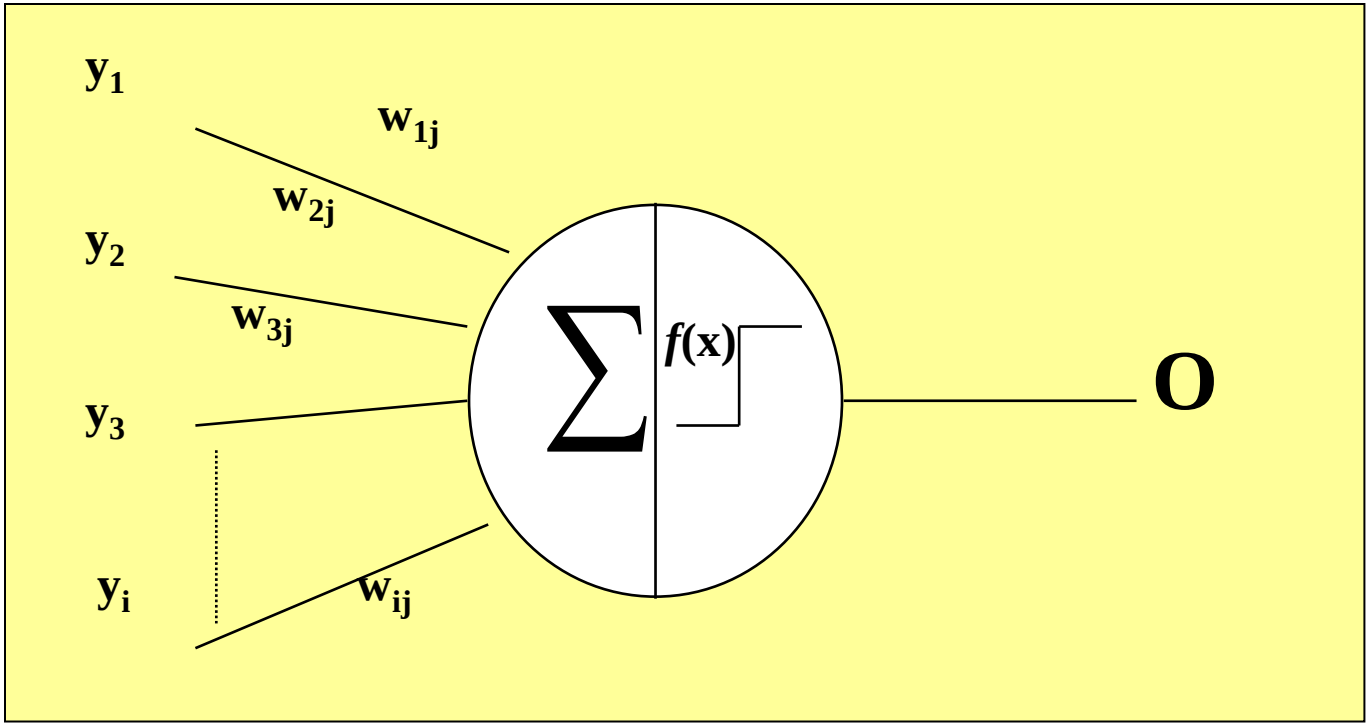
The Structure of Neurons

- A neuron only fires if its input signal exceeds a certain amount (the threshold) in a short time period.
- Synapses vary in strength
  - Good connections allowing a large signal
  - Slight connections allow only a weak signal
  - Synapses can be either excitatory or inhibitory.

The Artificial Neuron (Perceptron)



A Simple Model of a Neuron (Perceptron)

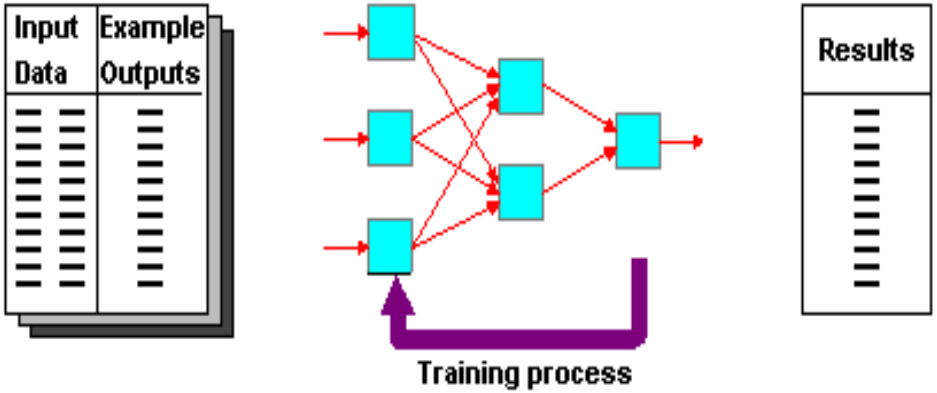


A Simple Model of a Neuron - how it works

- Each neuron has weighted inputs from other neurons and the input signals form a weighted sum
- Each neuron also has a threshold value and an activation function which transforms neuron's input into output as follows:
  - The unit performs a weighted sum of its inputs, and subtracts its threshold value, to give its activation level
  - Activation level is passed through a sigmoid activation function and if it exceeds the threshold, the neuron "fires", meaning that an output signal is given by the neuron

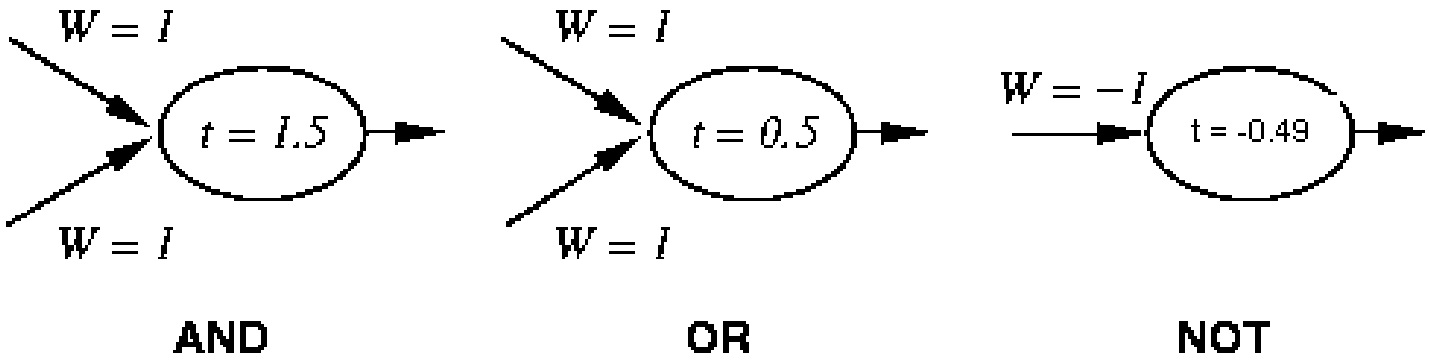
Supervised Learning

- Training and test data sets
- Training set; input & target



Sepal length	Sepal width	Petal length	Petal width	Class
5.1	3.5	1.4	0.2	0
4.9	3.0	1.4	0.2	2
4.7	3.2	1.3	0.2	0
4.6	3.1	1.5	0.2	1

Perceptron Training

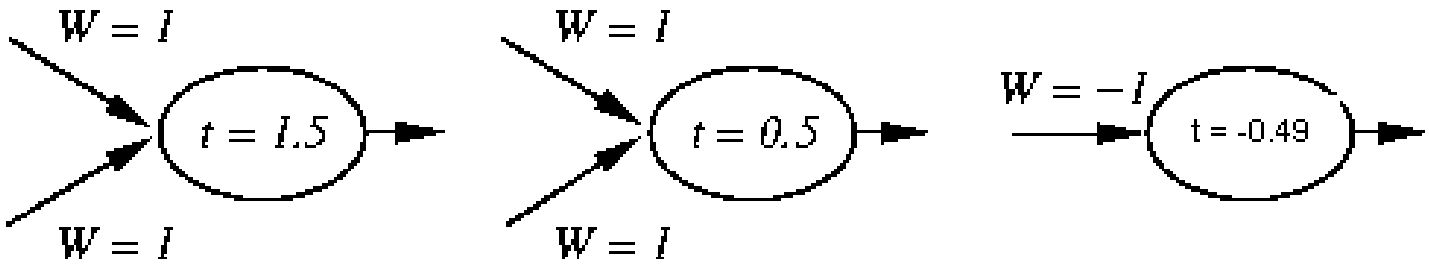


activation function

$$\text{Output} = \begin{cases} 1 \text{ if } \sum_{i=0} w_i x_i > t \\ 0 \text{ otherwise} \end{cases}$$

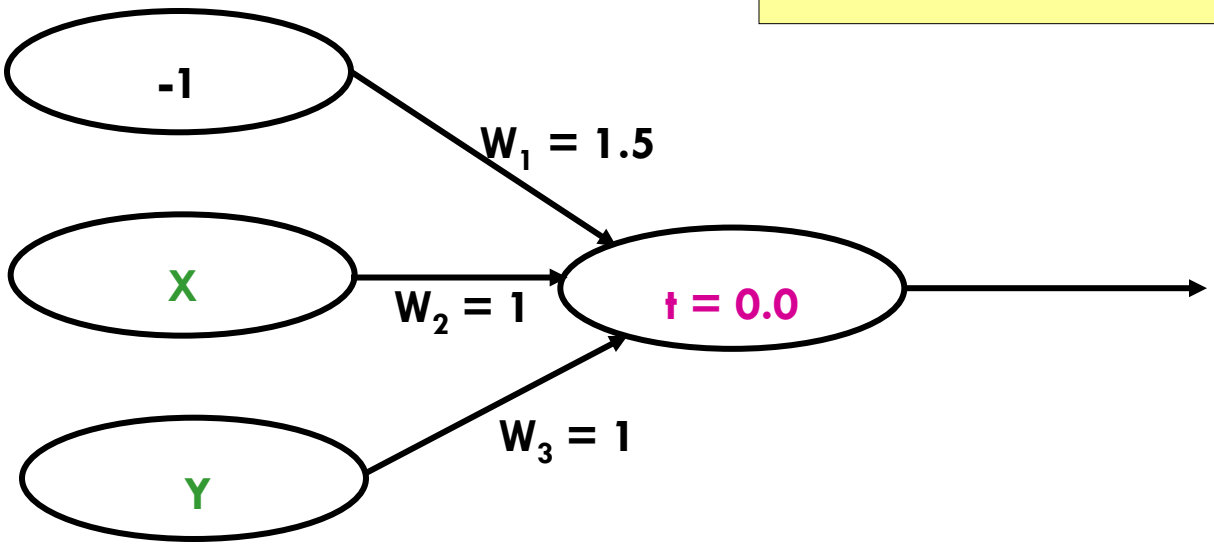
- Linear threshold is used.
- W - weight value
- t - threshold value

Simple network

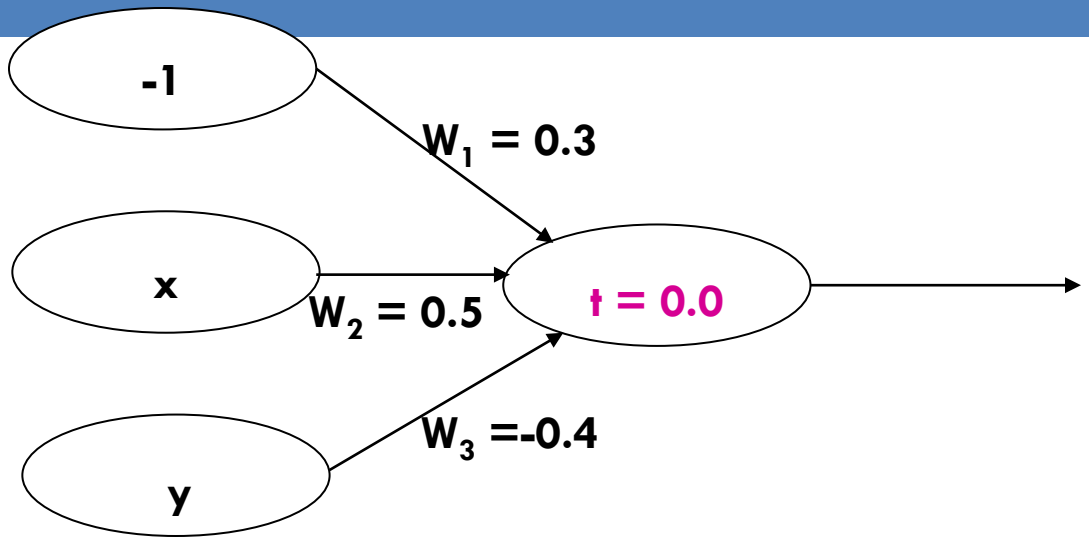


output= $\begin{cases} 1 & \text{if } \sum_{i=0} w_i x_i > t \\ 0 & \text{otherwise} \end{cases}$

AND with a Biased input



Training Perceptrons

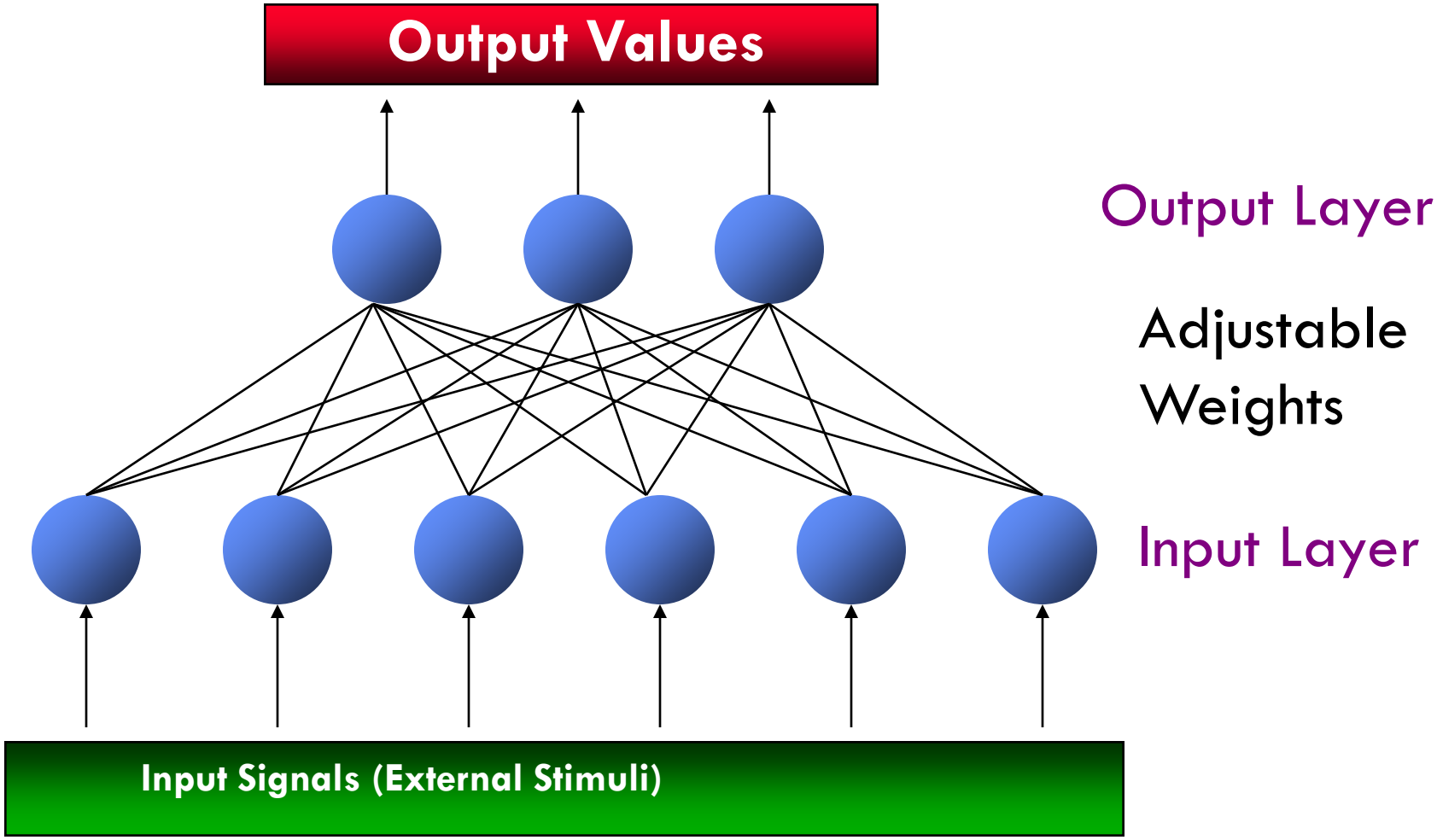


For AND

A	B	Output
0	0	0
0	1	0
1	0	0
1	1	1

I_1	I_2	I_3	Summation	Output
-1	0	0	(-1*0.3) + (0*0.5) + (0*-0.4) = -0.3	0
-1	0	1	(-1*0.3) + (0*0.5) + (1*-0.4) = -0.7	0
-1	1	0	(-1*0.3) + (1*0.5) + (0*-0.4) = 0.2	1
-1	1	1	(-1*0.3) + (1*0.5) + (1*-0.4) = -0.2	0

Multilayer Perceptron (MLP)



Types of Layers

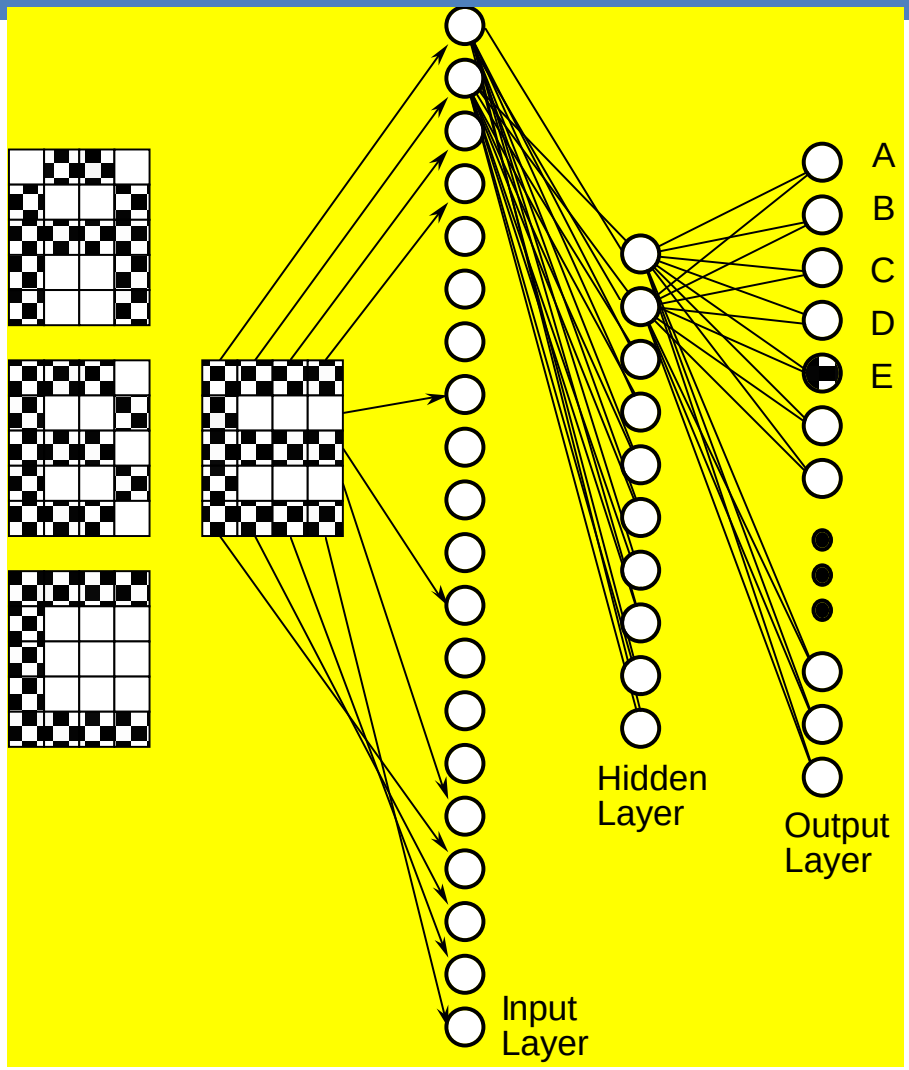
- The input layer.
  - Introduces input values into the network.
  - No activation function or other processing.
- The hidden layer(s).
  - Perform classification of features
  - Two hidden layers are sufficient to solve any problem
  - Features imply more layers may be better(There can be more than one hidden layers which are used for processing the inputs received from the input layers)
- The output layer.
  - Functionally just like the hidden layers
  - Outputs are passed on to the world outside the neural network.

Properties of neural networks

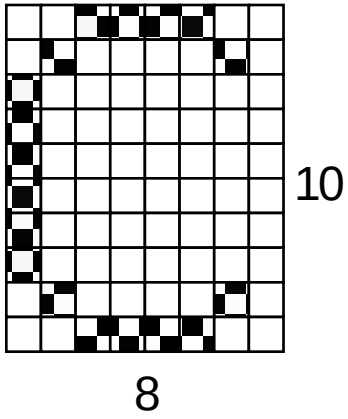
- The properties of neural networks define where they are useful.
  - Can learn complex mappings from inputs to outputs, based solely on samples
  - Difficult to analyse: firm predictions about neural network behaviour difficult;
    - Unsuitable for safety-critical applications.
  - Require limited understanding from trainer, who can be guided by heuristics.
  - **Application** - The properties of neural networks define where they are useful

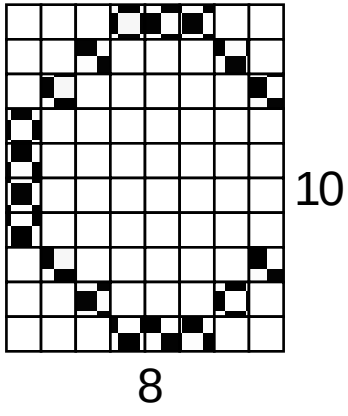
Neural network for OCR

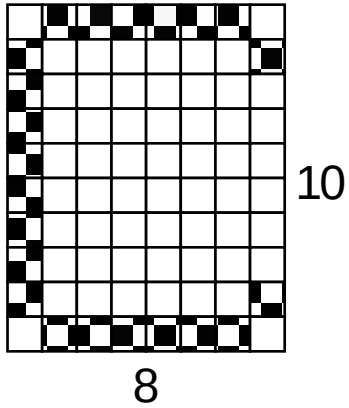
- OCR - optical character recognition
- feedforward network
- trained using Back-propagation



OCR for 8x10 characters

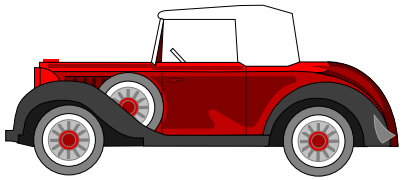






- NN are able to generalise
- learning involves generating a partitioning of the input space
- for single layer network input space must be linearly separable
- what is the dimension of this input space?
- how many points in the input space?
- this network is binary(uses binary values)
- networks may also be continuous

Engine management



- The behaviour of a car engine is influenced by a large number of parameters
  - temperature at various points
  - fuel/air mixture
  - lubricant viscosity.
- Major companies have used neural networks to dynamically tune an engine depending on current settings.

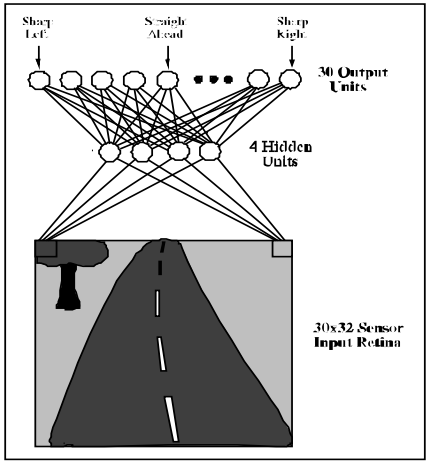
ALVINN

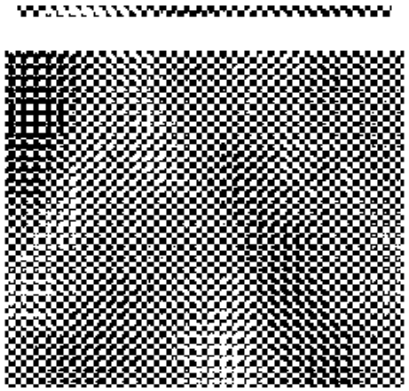
Drives 70 mph on a public highway

30 outputs for steering

4 hidden units

30x32 pixels as inputs



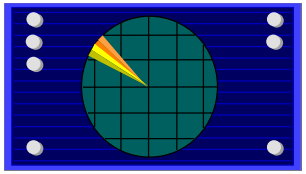


30x32 weights into one out of four hidden unit

Signature recognition

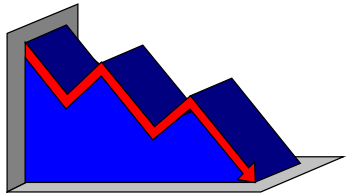
- Each person's signature is different.
- There are structural similarities which are difficult to quantify.
- One company has manufactured a machine which recognizes signatures to within a high level of accuracy.
  - Considers speed in addition to gross shape.
  - Makes forgery even more difficult.

Sonar target recognition



- Distinguish mines from rocks on sea-bed
- The neural network is provided with a large number of parameters which are extracted from the sonar signal.
- The training set consists of sets of signals from rocks and mines.

Stock market prediction



- “Technical trading” refers to trading based solely on known statistical parameters; e.g. previous price
- Neural networks have been used to attempt to predict changes in prices.
- Difficult to assess success since companies using these techniques are reluctant to disclose information.

Mortgage assessment

- Assess risk of lending to an individual.
- Difficult to decide on marginal cases.
- Neural networks have been trained to make decisions, based upon the opinions of expert underwriters.
- Neural network produced a 12% reduction in delinquencies compared with human experts.

Neural Network Problems

- Many Parameters to be set
- Overfitting
- long training times
- ...

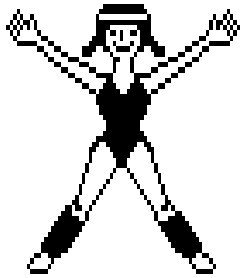
Parameter setting

- Number of layers
- Number of neurons
    - too many neurons, require more training time
- Learning rate
    - from experience, value should be small ~0.1
- Momentum term
- ..

Over-fitting

- With sufficient nodes can classify any training set exactly
- May have poor generalisation ability.
- Cross-validation with some patterns
  - Typically 30% of training patterns
  - Validation set error is checked each epoch
  - Stop training if validation error goes up

Training time



- How many epochs of training?
  - Stop if the error fails to improve (has reached a minimum)
  - Stop if the rate of improvement drops below a certain level
  - Stop if the error reaches an acceptable level
  - Stop when a certain number of epochs have passed